

Gyrovision research report

DANIELE GRAZIANI
MORPHEME CNRS/UNS/INRIA
I3s 2000 route des Lucioles BP 121 93 06903 SOPHIA ANTIPOLIS CEDEX, FRANCE
e.mail: graziani@i3s.unice.fr

LAURE BLANC-FÉRAUD
MORPHEME CNRS/UNS/INRIA
I3s 2000 route des Lucioles BP 121 06903 SOPHIA ANTIPOLIS CEDEX, FRANCE
e.mail: blancf@i3s.unice.fr

GILLES AUBERT
LABORATOIRE J.A. DIEUDONNÉ
Université de Nice SOPHIA ANTIPOLIS, Parc Valrose 06108 Nice CEDEX 2, FRANCE
e.mail: gaubert@unice.fr

Abstract

The report intends to give a, as much as possible, detailed description researches carried out by the CNRS, during the last two years, under Gyrovision the project In particular the present work is concerned with two different classical challenges in image processing, and with a possible solution strategy proposed by the authors. The first one is to develop a fast algorithm to process image of big size in the microscanning framework. The second one is to define, in the context of color demosaicking, a new method to perform at the same time: demosaicking, deblurring and denoising.

Introduction

The first part of the report is concerned with the classical image processing problem of reconstructing highly resolved images from several multiple less resolved images. Improvements in the resolution and fidelity of digital imaging systems have substantial value for remote sensing, military surveillance and other applications. Microscanning is a systematic approach to acquiring images with slightly different sample scene phases; between successive images the system is shifted slightly in a pre-determined controlled pattern. This makes an important difference with respect to general supersolution framework, where the system is shifted in a random pattern. First of all we describe the image formation model, by providing definitions of each standard acquisition operator: shifting, blurring, downsampling. Then in line with recent works (see, for instance [8, 16]), we introduce a convex energy used to restore highly resolved image out from low resolution frames. Such energy is made up by TV regularization term and L^2 -discrepancy term, which takes into account the acquisition process. Then we focus on the main issue of this first part, that is to address real or at least acceptable computation time. For instance in aerosurveillance the imaging system is embedded on board, making crucial to restore acquired images in real time. Several numerical strategies to perform efficiently total variation minimization have been proposed in the literature. In this direction, as a preliminary step, we adapted to microscanning setting, the approach proposed in [14].

However, because of its iterative-sequential formulation, this method is not able to address in real-time, or at least in an acceptable computational time, extremely large problems. For such large scale simulations we need to address methods which allow us to reduce the problem to a finite sequence of sub-problems of a more manageable size. To do this we adapt to our setting the overlapping domain decomposition algorithm for total variation minimization proposed in [9].

So that we are able to reduce the minimization of the energy to a finite sequence of sub-problems of a smaller size, so allowing, at least in principle, for parallel computation. Finally we show applications and results on real and synthetic data.¹

The second part of the report addresses another classical challenging problem in image processing: the so called demosaicking, used in digital cameras for reconstructing color images. Let us give a short description.

Most digital cameras use a single sensor which is placed in front of color filter array: the Bayer Matrix. The sensor therefore sampled only one color per spatial position and the

¹a version of the Matlab code is available at the I3S laboratory (CNRS/UNS)

observed image is degraded by the effect of mosaic generation. It is therefore necessary to implement, possibly fast, algorithms to define an image with three color components by spatial position. The set of techniques used in the literature, to resolve this problem, is huge. Without claiming of being exhaustive we refer the reader to [1] for a general dissertation.

The originality of our research, in this context, is to define a variational method well suited to take into account all possible degradation effects due to: mosaic effect, blur and noise. Looking at the literature in this direction it is worth mention the work of Condat (see [4]) where a demosaicking-denoising method is proposed, but without taking into account blur effects. In [7, 15] all the degradation effects are considered, but with regularization energies, which does not allow for fast convex optimization technique. Indeed in these works, in order to take into account the correlation between the RGB components, the prior regularization term has a complicated expression.

Here we analyze and test a new method to perform in a more direct and possibly faster way demosaicking-deblurring-denoising problem. Our approach is based on two steps. The first one, as in [4], is working in a suitable basis where the three color components are statistically decorrelated. Then we are able to write our problem as a convex minimization problem. Finally to solve such a problem and to restore the image, we adapt to our framework the, well known, ADMM (Alternating Direction Multipliers Minimization) convex optimization technique. The ADMM method and its variants are largely used to solve convex minimization problems in image processing. We refer the reader to [6], for a general dissertation on convex optimization techniques, such as ADMM methods or others, and their applications to image processing.

Organization of the report

The report is organized as follows. In section 1 of the first chapter we introduce the microscanning framework and we define the acquisition operator in this setting. In section 2 we illustrate the results obtained by adapting the Nesterov's strategy. In section 3 we illustrate the overlapping decomposition method for microscanning framework. Section 3 is completely devoted to run numerical tests.

In section 1 of chapter 2 we give a short description of the general ADDM method. In section 2 we define the new basis for which the channels of the color image are decorrelated. Section 3 is devoted to introduce the Bayer matrix and the blur operator. Section 4 is concerned with the definition of our variational model. In section 5 we show how to adapt the classical ADMM algorithm to our case.

Finally in the last section we give some applications of our algorithm on color images of big sizes.²

²a version of the matlab code is available at the I3S laboratory (CNRS/UNS)

Discrete setting

We define the discrete rectangular domain Ω of step size $\delta x = 1$ and dimension $d_1 d_2$. $\Omega = \{1, \dots, d_1\} \times \{1, \dots, d_2\} \subset \mathbb{Z}^2$. In order to simplify the notations we set $X = \mathbb{R}^{d_1 \times d_2}$ and $Y = X \times X$. $u \in X$ denotes a matrix of size $d_1 \times d_2$. For $u \in X$, $u_{i,j}$ denotes its (i, j) -th component, with $(i, j) \in \{1, \dots, d_1\} \times \{1, \dots, d_2\}$. For $g \in Y$, $g_{i,j}$ denotes the (i, j) -th component of g with $g_{i,j} = (g_{i,j}^1, g_{i,j}^2)$ and $(i, j) \in \{1, \dots, d_1\} \times \{1, \dots, d_2\}$. We endowed the space X and Y with standard scalar product and standard norm. For $u, v \in X$:

$$\langle u, v \rangle_X = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} u_{i,j} v_{i,j}.$$

For $g, h \in Y$:

$$\langle g, h \rangle_Y = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \sum_{l=1}^2 g_{i,j}^l h_{i,j}^l.$$

For $u \in X$ and $p \in [1, +\infty)$ we set:

$$\|u\|_p := \left(\sum_{i=1}^{d_1} \sum_{j=1}^{d_2} |u_{i,j}|^p \right)^{\frac{1}{p}}.$$

For $g \in Y$ and $p \in [1, +\infty)$:

$$\|g\|_p := \left(\sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \sum_{l=1}^2 |g_{i,j}^l|^p \right)^{\frac{1}{p}}.$$

If G, F are two vector spaces and $H : G \rightarrow F$ is a linear operator the norm of H is defined by

$$\|H\| := \max_{\|u\|_G \leq 1} (\|Hu\|_F).$$

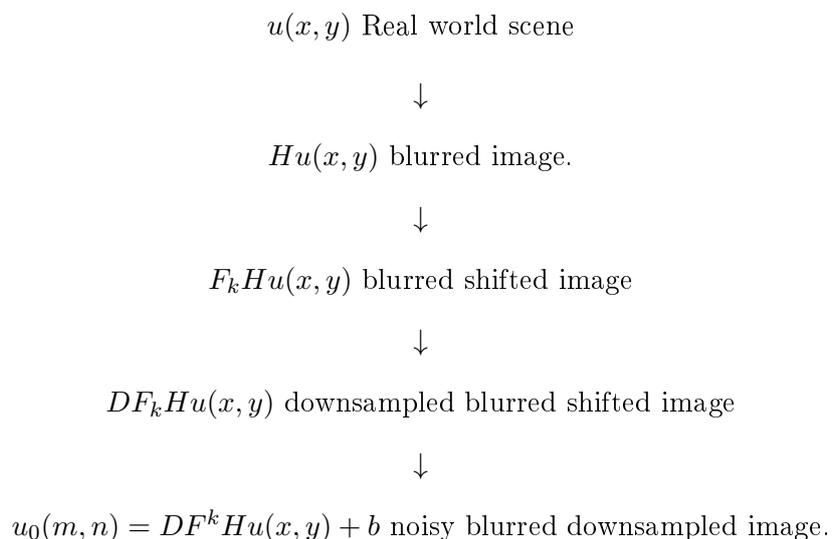
Chapter 1

Microscanning

We investigate a convex variational framework to reconstruct high resolution images from a low resolution video. We analyze the image formation process and provide a well designed model for warping, blurring, downsampling and restoration. We provide an investigation of each model components. The super resolution problem is modeled as a convex minimization problem. The minimization is performed with a recent decomposition domain technique, which allows for parallel computing and real time algorithm.

1.1 The model

In what follows HR and LR stand for high and low resolution respectively. As in [8] we assume the following acquisition process:



The super-solution reconstruction problem is the following:

Given a set $\{u_k^0\}_1^K$ of $K = r^2$ LR images (where r^2 is the resolution enhancement factor between the LR and the HR image), find u .

For any k we have the

$$u_k^0 = DF_kHu + b_k$$

where:

- u_k^0 is the LR frame: a vector of size $[MN \times 1]$
- D is the down-sampling operator: a matrix of size $[MN \times r^2MN]$
- H^k is the PSF: a matrix of size $[r^2MN \times r^2MN]$
- F^k are the shifting operators: each F_k is a matrix of size $[r^2MN \times r^2MN]$
- u is HR image: a vector of size: $[r^2MN \times 1]$.

Considering that the frames are acquired with a unique camera we assume the following fact: $D_k = D$ for each frame.

1.1.1 Assumption on the blur operator H

We define the blur operator H by means of its Fourier transform h : if f denotes a frequency in the Fourier space and f_c is the cutting frequency of the acquisition system.

$$(1.1) \quad h(f) = \begin{cases} \left| \frac{2}{\pi} x \left(\arcsin\left(\frac{f}{f_c} - \frac{x}{\gamma} \sqrt{1 - \left(\frac{f}{f_c}\right)^2}\right) \right) \right| & \text{if } f \leq f_c \\ 0 & \text{if } f \geq f_c \end{cases}$$

In particular we have that $\|h\|_\infty \leq 1$, which ensure that $\|H\| \leq 1$. Such a blur is often used in aerosurveillance system camera.

1.1.2 Assumption on the downsampling operator

Given a continuous image u , we define the pixel value $u_{i,j}$ of the corresponding discrete image at the position (i, j) by integration of u on the area $\Delta_{i,j} = (i, j) + [-\frac{1}{2}, \frac{1}{2}]^2$. Then we define the downsampling operator as:

$$u_{i,j} \mapsto Du_{i,j} = u_{k,l} = \frac{1}{2r^2} \frac{1}{A(\Delta_{k,l}^r)} \sum_{0 \leq i,j \leq MN} A(\Delta_{i,j} \cap \Delta_{k,l}^r) u_{i,j},$$

where $\Delta_{k,l}^r = (k, l) + [-\frac{r}{2}, \frac{r}{2}]^2$ and r is the scale factor and $1 \leq k, l \leq \frac{M}{r}, \frac{N}{r}$. A is the common Lebesgue measure.

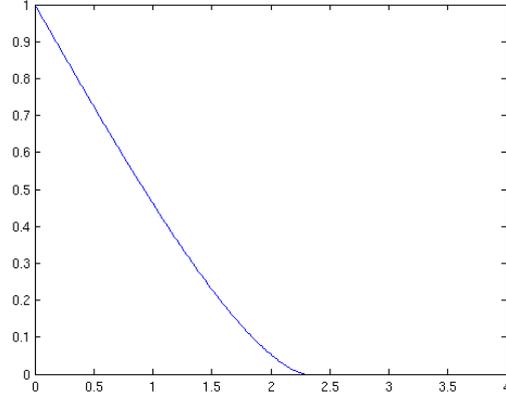


Figure 1.1: The profile of the Fourier transform of the blur operator with cutting frequency $f_c = 2.27$.

It is not difficult to check that $\|D\| \leq \frac{1}{2r^2} < \frac{1}{r^2}$.

Remark 1.1 Let us set $T_k = DF_kH$. By the assumptions on the acquisition operators we get the following bound:

$$(1.2) \quad \left\| \sum_{k=1}^{r^2} T_k \right\| \leq \sum_{k=1}^{r^2} \|T_k\| \leq \sum_{k=1}^{r^2} \|D\| \|F^k\| \|H\| \leq r^2 \frac{1}{2r^2} < 1.$$

1.2 Nesterov's strategy

To retrieve u we minimize the residual error between the model and the measurements:

$$\min_u \sum_{k=1, \dots, K} \|T_k u - u_k^0\|_2^2,$$

with $K = r^2$. This minimization problem is ill-posed therefore we need to add an a priori information on the unknown u . Then we are led to the TV-super resolution problem:

$$(1.3) \quad \min_u \lambda |\nabla u|(\Omega) + \sum_{k=1, \dots, K} \|T_k u - u_k^0\|_2^2,$$

where λ is a positive weight and, in the discrete setting, $|\nabla u|(\Omega)$ is the total variation on Ω of u , which in discrete setting is simply given by the l^1 -norm of the ∇u . Here we briefly recall the fast descent gradient Nesterov's algorithm (see [11, 12]). We state it in the formulation proposed in [13, 14]. For further details and general statements we refer the reader to [13, 14] and references therein.

Definition 1.1 Let $\psi : X \rightarrow \mathbb{R}$ be a convex function. The operator

$$\text{prox}_\psi : X \rightarrow X \quad x \mapsto \arg \min_{y \in X} \left\{ \psi(y) + \frac{1}{2} \|y - x\|_2^2 \right\}$$

is called proximal operator associated to ψ .

If $\text{prox}_{\lambda\psi}$ can be computed exactly for every $\lambda \geq 0$ and every $x \in X$, the function ψ is said to be simple.

Proposition 1.1 Let $F : X \rightarrow \mathbb{R}$ be given by:

$$F(u) = F_1(u) + F_2(u) \quad \text{for } u \in X,$$

where F_1 is a convex L -Lipschitz differentiable function and F_2 a simple function. Then the following algorithm¹:

$$(1.4) \quad \left\{ \begin{array}{l} u_0 \in X \quad A_0 = 0 \quad g = 0 \quad u = 0 \\ \text{do for } l : 1, \dots, L_{max} \\ t = \frac{2}{L} \\ a = t + \sqrt{t^2 + 4tA} \\ v = \text{prox}_{AF_2}(u_0 - g) \\ y = \frac{Au + Av}{A+a} \\ u = \text{prox}_{\frac{1}{L}F_2}(y - \frac{1}{L}\nabla F_1(y)) \\ g = g + a\nabla F_1(u) \\ A = A + a \end{array} \right.$$

ensures that:

$$(1.5) \quad 0 \leq F(u_l) - F(u^*) \leq L \frac{|u^* - u_0|_2}{l^2},$$

where $u^* \in X$ is a minimum point of F and $u_0 \in X$ is an initial data.

The iteration numbers L_{max} is fixed via inequality (1.5). For instance, if we want an error of order $\frac{1}{l^2}$, we need then a number L_{max} of iterations of order l .

In order to apply the previous algorithm we set $F_1(u) = \sqrt{|\nabla u|^2 + \epsilon} + \sum_{k=1, \dots, K} \|T_k u - u_k^0\|_2^2$ with ϵ small parameter and $F_2 = 0$.

We made run the Matlab code on an Intel(R) Xeon(R) CPU 5120 @ 1.86GHz.

1.3 The Overlapping Domain decomposition method

To retrieve the super resolution image u we wish to minimize the following energy:

$$(1.6) \quad \mathcal{F}(u) = \sum_{k=1}^{r^2} \|T^k u - u_0^k\| + 2\lambda |\nabla u|(\Omega)$$

where $T^k = DHF^k$ is a linear operator belonging to $\mathcal{L}(\mathbb{R}^{MN}, \mathbb{R}^{\frac{MN}{r^2}})$ and $|\nabla u|(\Omega)$ stands for the total variation of u in Ω . In order to obtain a fast minimization we follow the decomposition

¹we omit the dependence on l to simplify the notation

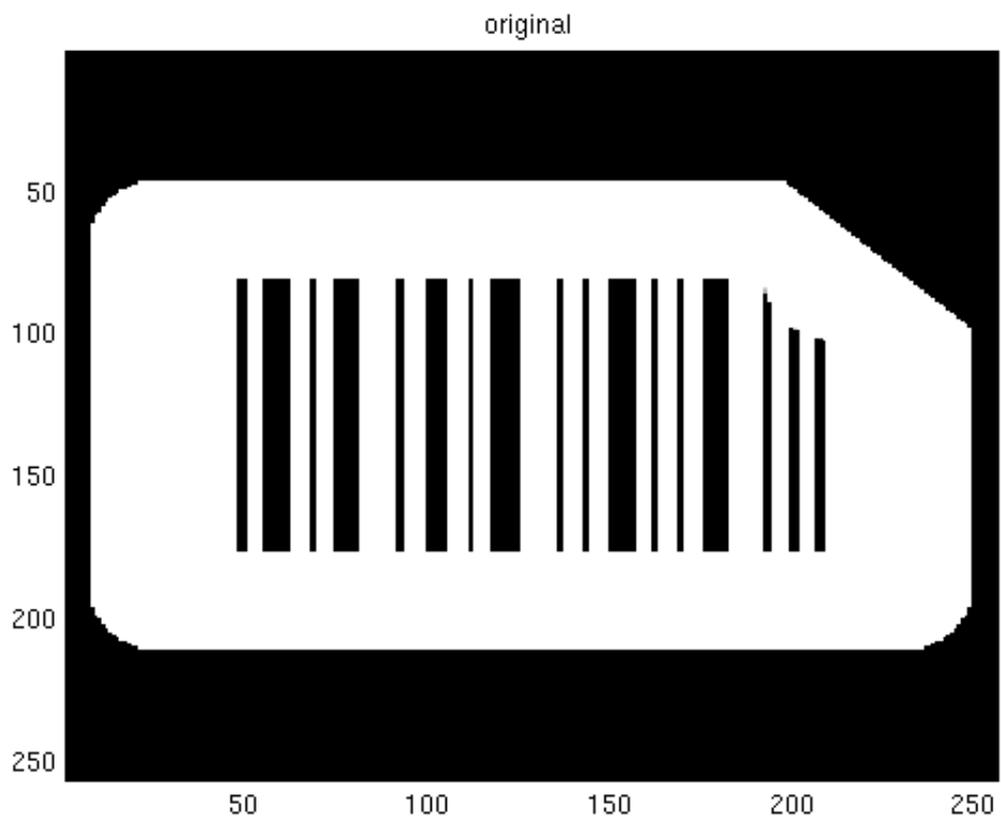


Figure 1.2: *original image.*

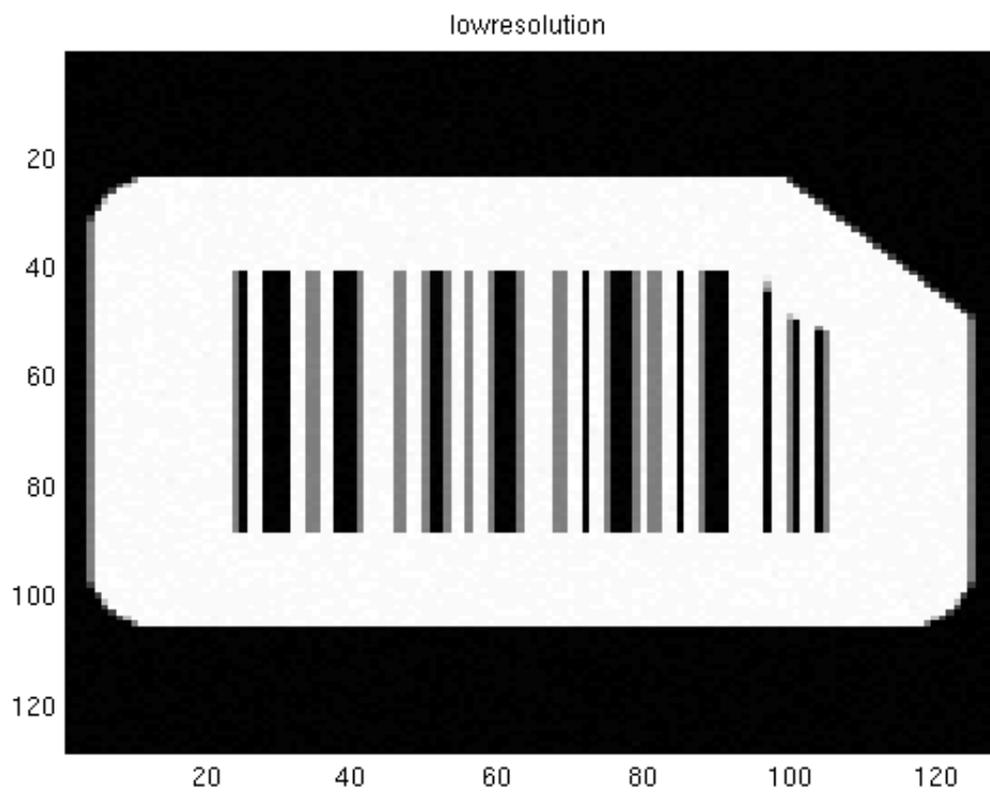


Figure 1.3: *downsampled LR image $r^2 = 4$.*

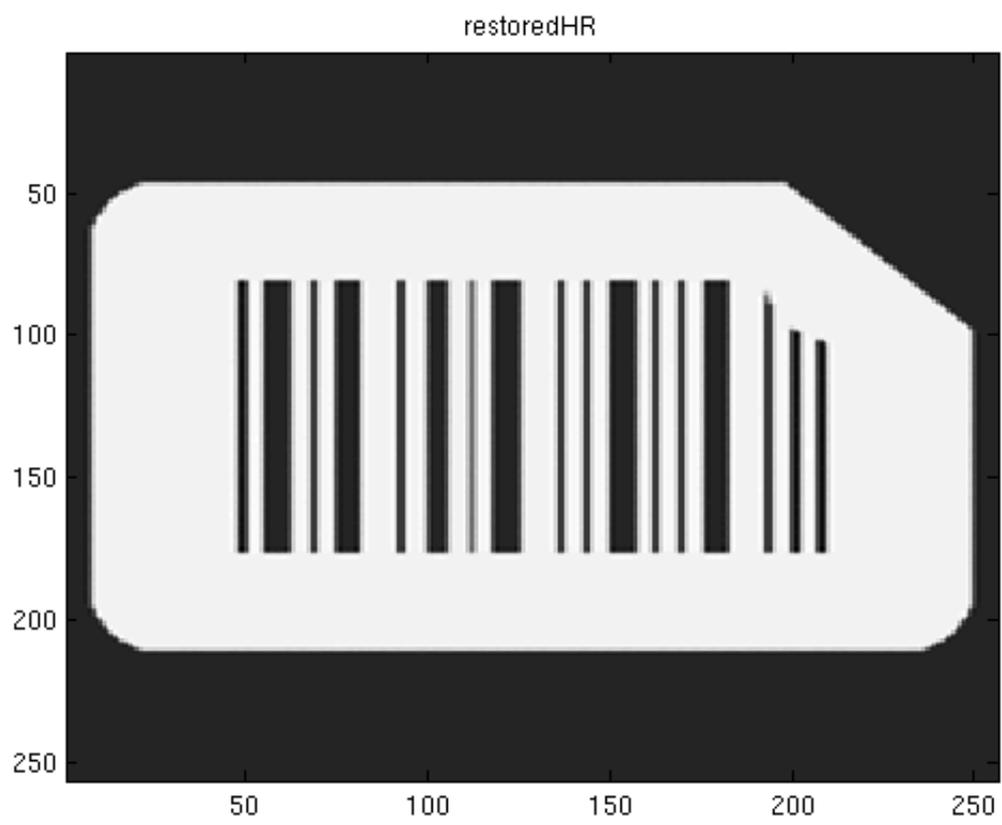


Figure 1.4: *Restored HR image.*

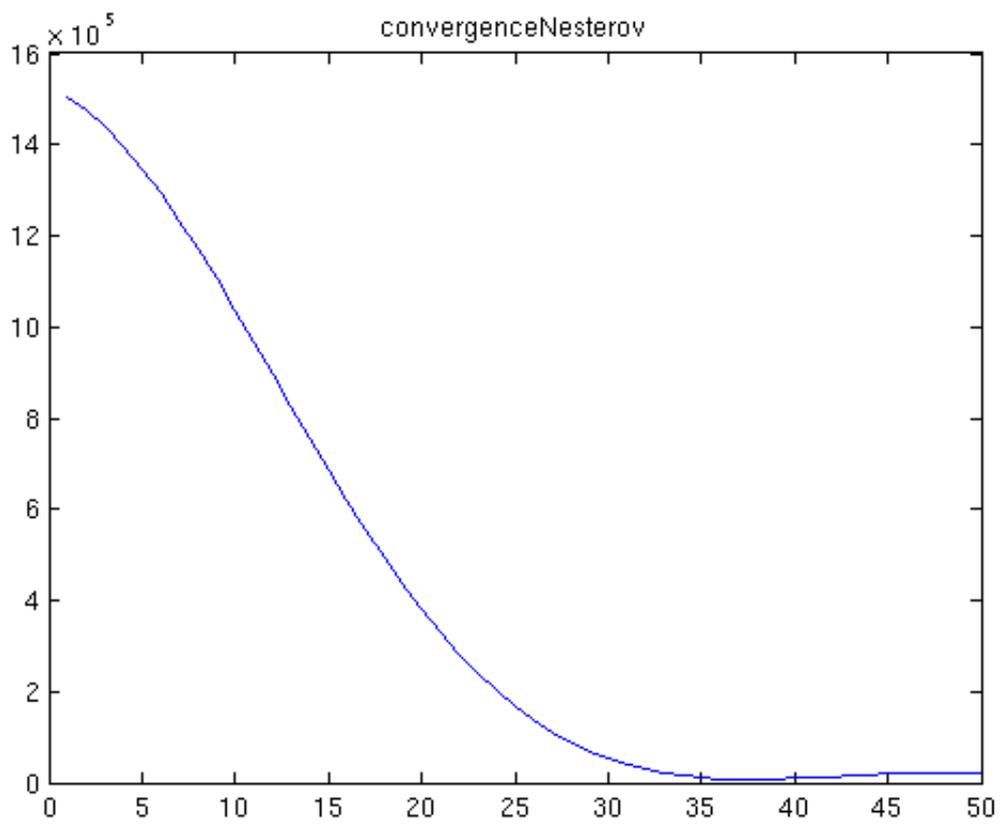


Figure 1.5: *Convergence curve of algorithm 1.4. Size image 256x256. CPU time 5mn.*



Figure 1.6: The domain decomposition. The blue line is the interface Γ_1 . The red line is the interface Γ_2

overlapping domain method used in [9, 10]. So that instead of minimizing (1.6) on the whole image domain, we split Ω in two overlapping sub domains Ω_1 and Ω_2 such that $\Omega_1 \cap \Omega_2 \neq \emptyset$.

We denote by Γ_1 the interface between Ω_1 and $\Omega_2 \setminus \Omega_1$ and by Γ_2 the interface between Ω_2 and $\Omega_1 \setminus \Omega_2$.

\mathbb{R}^{MN} is decomposed in two closed subspaces $V_j = \{u \in \mathbb{R}^{MN} : \text{supp}(u) \subset \Omega_j\}$. Then we can rewrite any $u \in \mathbb{R}^{MN}$ as

$$\begin{cases} u_1(x) & x \in \Omega_1 \setminus \Omega_2 \\ u_1(x) + u_2(x) & x \in \Omega_1 \cap \Omega_2 \\ u_2(x) & x \in \Omega_2 \setminus \Omega_1 \end{cases}$$

We define the trace operator as

$$Tr_{\lfloor \Gamma_i} : V_i \mapsto \mathbb{R}^{\Gamma_i}, \quad i = 1, 2,$$

with $Tr_{\lfloor \Gamma_i} v_i = v_{\lfloor \Gamma_i}$. We also require the following splitting properties of the total variation:

$$(1.7) \quad |\nabla u|(\Omega) = |\nabla u_{\lfloor \Omega_1}|(\Omega_1) + c_1(u_{\lfloor (\Omega_2 \setminus \Omega_1) \cup \Gamma_1})$$

$$(1.8) \quad |\nabla u|(\Omega) = |\nabla u_{\lfloor \Omega_2}|(\Omega_2) + c_1(u_{\lfloor (\Omega_1 \setminus \Omega_2) \cup \Gamma_2}),$$

with c_1, c_2 suitable functions (see [9]) Finally to control the solutions on the overlapping parts one needs to fix a bounded uniform partition of unity, that is $\{\chi_1, \chi_2\}$ such that

1. $Tr_{\lfloor \Gamma_i} \chi_i = \chi_{\lfloor \Gamma_i} = 0$ for $i = 1, 2$,

2. $\chi_1 + \chi_2 = 1$,
3. $\text{supp}\chi_i \subset \Omega_i$ for $i = 1, 2$,
4. $\max\{\|\chi_1\|_\infty, \|\chi_2\|_\infty\} = c < +\infty$.

One wishes to minimize energy (1.6) by picking an initial data $u^0 = \tilde{u}_1^0 + \tilde{u}_2^0 \in V_1 + V_2$ and iterate the following procedure:

$$(1.9) \quad \begin{cases} u_1^{n+1} \cong \operatorname{argmin}_{v_1 \in V_{1Tr|_{\Gamma_1}} v_1=0} \mathcal{F}(v_1 + \tilde{u}_2^n) \\ u_2^{n+1} \cong \operatorname{argmin}_{v_2 \in V_{2Tr|_{\Gamma_1}} v_2=0} \mathcal{F}(u_1^{n+1} + v_2) \\ u^{n+1} := u_1^{n+1} + u_2^{n+1} \\ \tilde{u}_1^n := \chi_1 u^{n+1} \\ \tilde{u}_2^n := \chi_2 u^{n+1} \end{cases}$$

1.3.1 Subspace minimization

Let us consider the minimization on Ω_1 . The minimum problem is the following:

$$(1.10) \quad \begin{aligned} & \operatorname{argmin}_{v_1 \in V_{1Tr|_{\Gamma_1}} v_1=0} \mathcal{F}(v_1 + u_2) \\ &= \operatorname{argmin}_{v_1 \in V_{1Tr|_{\Gamma_1}} v_1=0} \sum_{k=1}^{r^2} \|T^k v_1 - (u_0^k - T^k u_2)\|_2^2 + 2\lambda |\nabla(v_1 + u_2|_{\Omega_1})|(\Omega). \end{aligned}$$

As in [9] we introduce the surrogate functional to separate the variable u_1 from the action of the operators T^k . For $a, u_1 \in V_1, u_2 \in V_2$ we define

$$\mathcal{F}_1^s(u_1 + u_2, a) := \mathcal{F}(u_1 + u_2) + r^2 \|u_1 - a\|_2^2 - \sum_{k=1}^{r^2} \|T^k(u_1 - a)\|_2^2.$$

By same computation of [9] we obtain that

$$\mathcal{F}_1^s(u_1 + u_2, a) := \sum_{k=1}^{r^2} \|u_1 - (a + (T^{k*}(u_0^k - T^k u_2 - T^k a))|_{\Omega_1})\|_2^2 + 2\lambda |\nabla(u_1 + u_2)|(\Omega) + \Phi(a, u_0^k, u_2),$$

where Φ does not depend on u_1 . Then we obtain an approximate solution of problem (1.10) by using the following algorithm:

$$(1.11) \quad \begin{cases} u_1^{l+1} = \operatorname{argmin}_{u_1 \in V_{1Tr|_{\Gamma_1}} u_1=0} \mathcal{F}_1^s(u_1 + u_2, u_1^l), & l \geq 0 \\ u_1^0 = \tilde{u}_1^0 \in V_1 \end{cases} .$$

Moreover in (1.11) the total variation can be restricted on Ω_1 only, since, by using (1.7) and the interior boundary condition we have

$$(1.12) \quad |\nabla(u_1 + u_2)|(\Omega) = |\nabla(u_1 + u_2)|_{[\Omega_1]} |(\Omega_1) + c_1(u|_{[(\Omega_2 \setminus \Omega_1) \cup \Gamma_1]}).$$

Hence problem (1.11) turns out to be equivalent to:

$$\operatorname{argmin}_{u_1 \in V_1 Tr_{[\Gamma_1]} u_1 = 0} \sum_{k=1}^{r^2} \|z_1^k - u_1\|_2^2 + 2\lambda |\nabla(u_1 + u_2)|_{[\Omega_1]} |(\Omega_1),$$

where $z_1^k = u_1^l + (T^{k*}(u_0^k - T^k u_2 - T^k u_1^l))|_{[\Omega_1]}$.

1.3.2 Oblique thesholding

We give here the analogous of Theorem 4.3 of [9]. The proof is the same up to minor changes.

Theorem 1.1 *For $u_2 \in V_2$, $z_1^k \in V_1$ and $k = 1, \dots, r^2$ the following statements are equivalent:*

1. $u_1^* = \operatorname{argmin}_{u_1 \in V_1 Tr_{[\Gamma_1]} u_1 = 0} \sum_{k=1}^{r^2} \|z_1^k - u_1\|_2^2 + 2\lambda |\nabla(u_1 + u_2)|_{[\Omega_1]} |(\Omega_1)$,
2. *there exists $\eta \in V_1$ with $\operatorname{supp} \eta = \Gamma_1$ such that $\eta = (Tr_{[\Gamma_1]})^* Tr_{[\Gamma_1]} (\sum_{k=1}^{r^2} z_1^k + P_{\lambda K}(\eta - (\sum_{k=1}^{r^2} z_1^k + u_2)))$.*

where $K = \{\operatorname{div} p : \|p(x)\|_\infty \leq 1 \forall x \in \Omega\}$.

The next proposition allows for explicit computation of η by iterating the following procedure:

$$(1.13) \quad \begin{cases} \eta^{(0)} \in V_1, & \operatorname{supp} \eta^0 = \Gamma_1 \\ \eta^{m+1} = (Tr_{[\Gamma_1]})^* Tr_{[\Gamma_1]} (\sum_{k=1}^{r^2} z_1^k + P_{\lambda K}(\eta^m - (\sum_{k=1}^{r^2} z_1^k + u_2))) \end{cases}$$

Proposition 1.2 *The following statements are equivalent:*

1. *there exists $\eta \in V_1$ such that $\eta = (Tr_{[\Gamma_1]})^* Tr_{[\Gamma_1]} (\sum_{k=1}^{r^2} z_1^k + P_{\lambda k}(\eta - (\sum_{k=1}^{r^2} z_1^k + u_2)))$*
2. *the iteration (1.13) converges to any η that satisfies condition (2) in Theorem 1.1.*

The proof is as in [9] Proposition 4.4.

1.3.3 Convergence of the subspace minimization and computation of u_1

We are now in position of computing iteration (1.11), which gives the solution of the subspace minimization problem. Indeed we can iterate as follows

$$(1.14) \quad u_1^{l+1} = (I - P_{\alpha K}) \left(\sum_{k=1}^{r^2} u_1^l + T^{k*}(u_0^k - T^k u_2 - T^k u_1^l) + u_2 - \eta^l \right) - u_2$$

where η^l is computed via iteration (1.13). Now (1.14) is equivalent, by recalling that $z_1^k = u_1^l + (T^{k*}(u_0^k - T^k u_2 - T^k u_1^l))|_{\Omega_1}$, to

$$(1.15) \quad u_1^{l+1} = (I - P_{\lambda K}) \left(\sum_{k=1}^{r^2} z_1^k + u_2 - \eta^l \right) - u_2$$

So that in order to compute u_1 we just need to know how to compute the projection $P_{\lambda K}$ onto λK , where $K = \{ \operatorname{div} p : \|p(x)\|_\infty \leq 1 \ \forall x \in \Omega \}$.

This can be done as in [3], where such a projection is related to the convex conjugate of the total variation.

One can compute $P_{\lambda K}(f)$ of some element f , as the limit of $\alpha \operatorname{div} p_n$ where p_n is obtain by iterating the following procedure:

$$\begin{cases} p(0) = 0 \\ p^{n+1} = \frac{p^n + \tau(\nabla(\operatorname{div} p^n - \frac{f}{\lambda}))}{1 + |\tau(\nabla(\operatorname{div} p^n - \frac{f}{\lambda}))|}. \end{cases}$$

Then if $\tau < \frac{1}{8}$, one gets that $\lambda \operatorname{div} p_n \rightarrow P_{\lambda K}(f)$.

Proposition 1.3 *The iteration (1.14) converges to a solution $u_1^* \in V_1$ of (1.10) for any initial choice of u_1^0 .*

The proof of this proposition is as the proof of Proposition 4.7 of [9], since we have bound (1.2).

1.3.4 Convergence Properties

We go back to the sequential algorithm (1.9). We rewrite the algorithm as follows: Pick as initial data $u^0 = \tilde{u}_1^0 + \tilde{u}_2^0 \in V_1 + V_2$ and iterate:

$$(1.16) \quad \begin{cases} \begin{cases} u_1^{(n+1,0)} = \tilde{u}_1^n \\ u_1^{(n+1,l+1)} = \operatorname{argmin}_{u_1 \in V_1 T r|_{\Gamma_1=0}} = \mathcal{F}_1^s(u_1 + \tilde{u}_2^n, u_1^{(n+1,l)}) \quad l = 0, \dots, L-1 \end{cases} \\ \begin{cases} u_2^{(n+1,0)} = \tilde{u}_2^n \\ u_2^{(n+1,l+1)} = \operatorname{argmin}_{u_2 \in V_2 T r|_{\Gamma_2=0}} = \mathcal{F}_2^s(u_1^{(n+1,L)} + u_2^n, u_2^{(n+1,m)}) \quad m = 0, \dots, M-1 \end{cases} \\ u^{(n+1)} := u_1^{(n+1,L)} + u_2^{(n+1,M)} \\ \tilde{u}_1^{n+1} := \chi_1 u^{n+1} \\ \tilde{u}_2^{n+1} := \chi_2 u^{n+1} \end{cases}$$

By using the bound (1.2), we have as in [9] the following proposition (see Proposition 5.4 and Theorem 5.7 of [9]). The numbers of iterations L, M are chosen according to the desired

error in the computation of η with procedure 1.14. By error we mean the l^2 -distance between u_l and u_{l-1} in the subdomain Ω_1 , and between u_m and u_{m-1} in the subdomain Ω_2 .

Proposition 1.4 *The algorithm (1.16) produces a sequence (u^n) with the following properties:*

1. $\mathcal{F}(u^{(n)}) > \mathcal{F}(u^{(n+1)})$ for all $n \in \mathbb{N}$ (unless $u^{(n)} = u^{(n+1)}$)
2. $\lim_{n \rightarrow +\infty} \|u^{(n+1)} - u^{(n)}\|_2 = 0$;
3. the sequence $(u^{(n)})$ has convergent subsequences.
4. the accumulation points of the sequence $(u^{(n)})$ are minimizers of \mathcal{F} . If \mathcal{F} has a unique minimizer, then the sequence $(u^{(n)})$ converges to it.

1.4 Numerics

We simulate the super resolution problem. We stress out that in order to perform numerical scheme we suppose to have the SR image that we want to reconstruct. Clearly in the real cases one only disposes of the LR frames. We made run the Matlab code on an Intel(R) Xeon(R) CPU 5120 @ 1.86GHz. Here below the results.

1.4.1 4 frames

- Size of images 512x512. number of sub domains 4
- CPU in the sub domains about 11 sec.
- MAE (Mean Absolute Error) between the original SR-image and the restored image is $9.3712e - 04$
- MAE between the blurred image and the restored image is $21.3712e - 04$.

One can notice that the error between the restored image and the SR original image is smaller than the error between the blurred image and the restored image is $8.3712e - 04$. It means that the algorithm performs a correct deblurring.

1.4.2 16 frames

We perform the same experiment but with 16 LR-frames of size 64×64 . The LR-frames are obtained as before, by performing 16 shifting by using different motion vectors. The weight λ is fixed as 0.2. The computation time is about 2 s in the sub domains. Here below the results.

Real Image SR image



Figure 1.7: The SR real image we want to reconstruct.



Figure 1.8: The blurred image.



Figure 1.9: One of the 4 LR noisy and blurred frames. In the real case these frames are the only data we have.

Restored image



Figure 1.10: The restored SR image

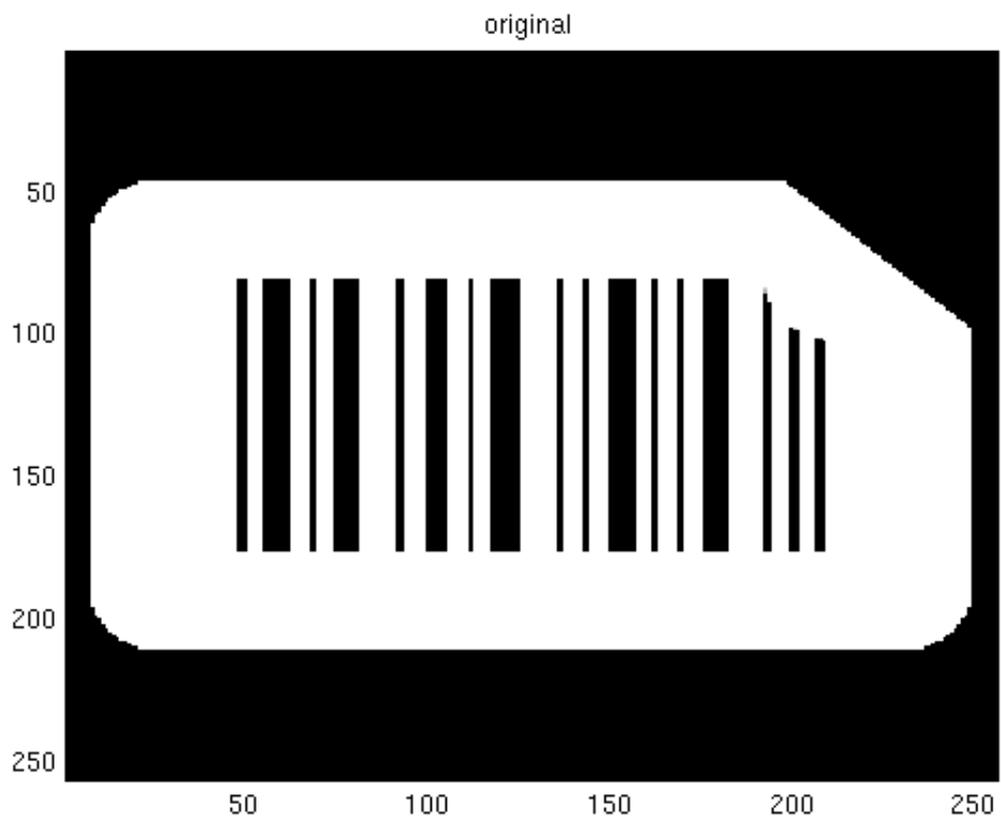


Figure 1.11: Original image

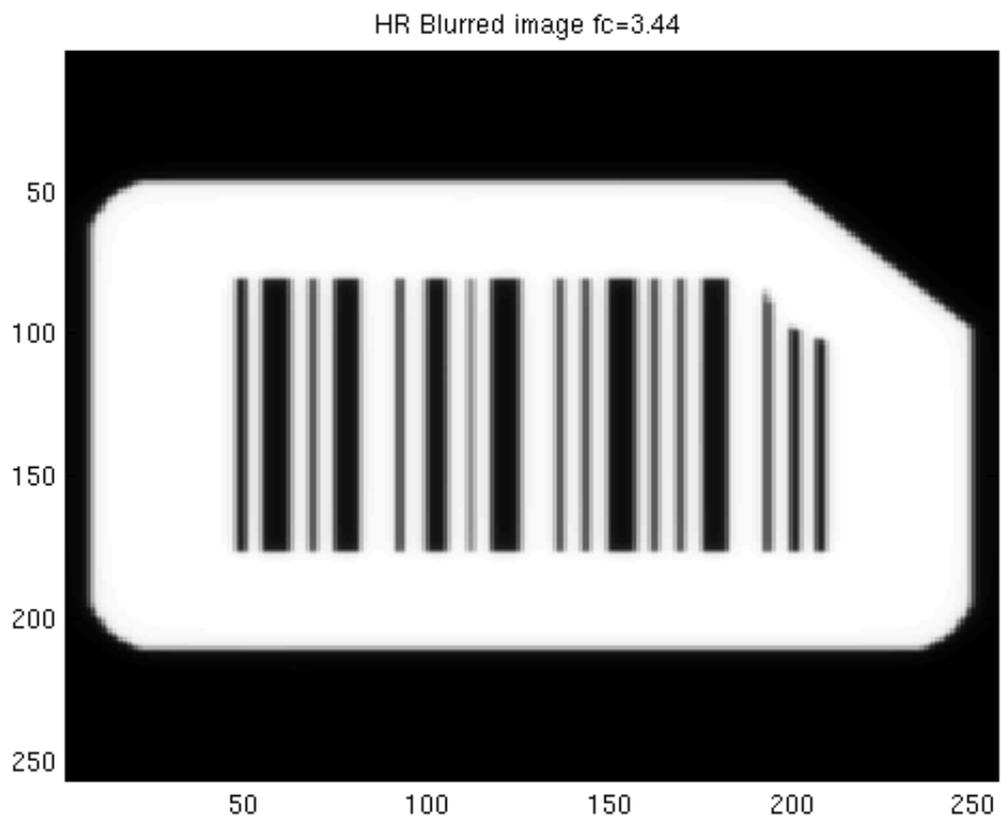


Figure 1.12: Blurred image with FTM threshold $f_C = 2.27$

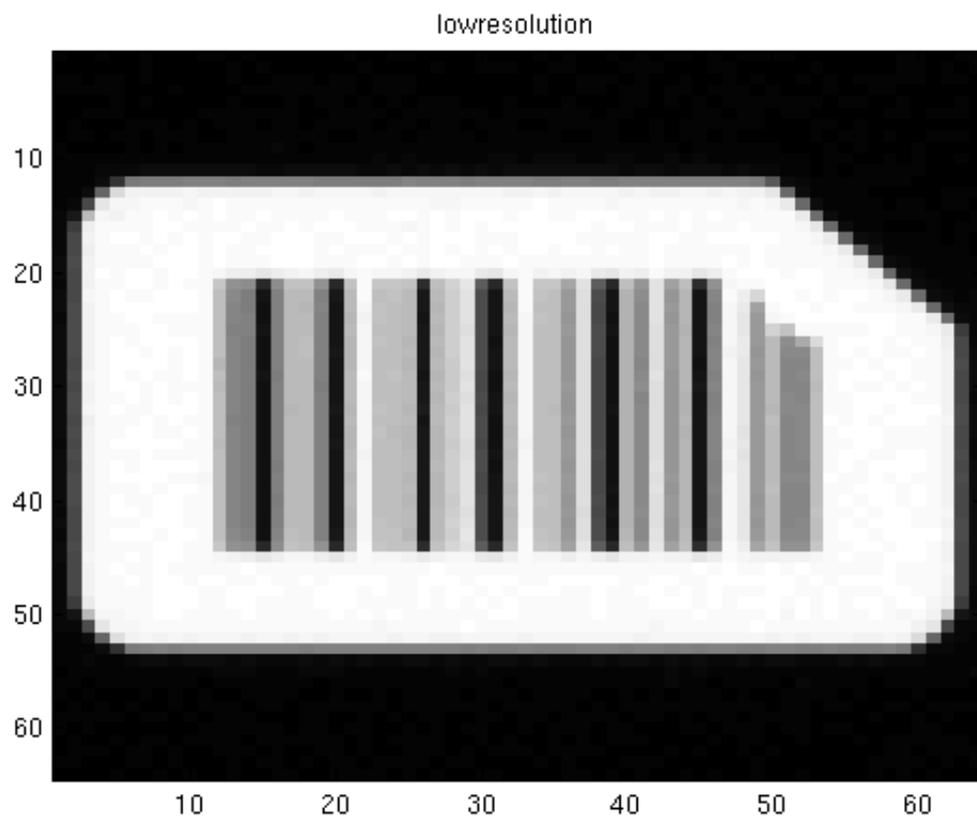


Figure 1.13: one of the 16 downsampled LR images with $r^2 = 4$.

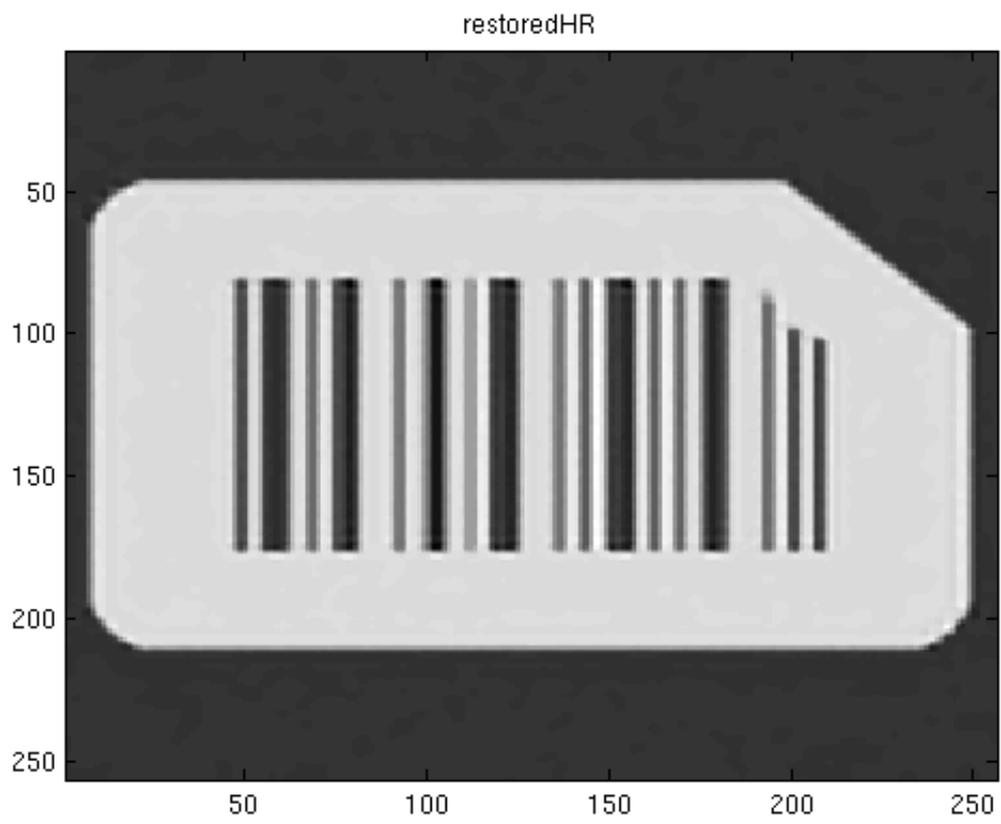


Figure 1.14: Restored HR image.

Chapter 2

Demosaicking-Deblurring-Denoising

We describe here a new algorithm to perform, in the same time, demosaicking deblurring, denoising. To this purpose we will adapt to our context an ADMM type algorithm. We recall the relevant features necessary to illustrate the application of such a method to our setting. We refer the reader to [6] and references therein, for a general dissertation on convex optimization techniques in image processing and recent developments on this matter.

2.1 ADMM algorithm for constrained minimization problem

In this section we describe the optimization method, we will adapt to our setting. The so called Alternating Direction Minimization Multipliers method ADMM. This particular optimization technique is well suited for constrained minimization problem of the following form:

$$(2.1) \quad \min_{u,z} F(z) + G(u) \quad \text{subject to } Bz + Au = b$$

where $F, G : \mathbb{R}^d \rightarrow \mathbb{R}$ and A and B matrix.

To solve problem (2.1) one considers the augmented Lagrangian and seeks its stationary points.

$$(2.2) \quad L_\alpha(z, u, \lambda) = F(z) + G(u) + \langle \lambda, Au + Bz - b \rangle + \frac{\alpha}{2} \|Au + Bz - b\|^2.$$

Then one iterate as follows:

$$(2.3) \quad \begin{cases} (z^{k+1}, u^{k+1}) = \operatorname{argmin}_{z,u} L_\alpha(z, u, \lambda^k) \\ \lambda^{k+1} = \lambda^k + \alpha(Au^{k+1} + Bz^{k+1} - b), \quad \lambda^0 = 0 \end{cases}$$

The following result has been proven in [5].

Theorem 2.1 (Eckstein, Bertsekas) *Suppose B has full column rank and $G(u) + \|A(u)\|^2$ is strictly convex. Let λ_0 and u_0 arbitrary and let $\alpha > 0$. Suppose we are also given sequences $\{\mu_k\}$ and $\{\nu_k\}$ with $\sum_k \mu_k < \infty$ and $\sum_k \nu_k < \infty$. Assume that*

1. $\|z^{k+1} - \operatorname{argmin}_{z \in \mathbb{R}^N} F(z) + \langle \lambda^k, Bz \rangle + \frac{\alpha}{2} \|Au^k + Bz - b\|^2\| \leq \mu_k$

$$2. \|u^{k+1} - \operatorname{argmin}_{z \in \mathbb{R}^M} G(u) + \langle \lambda^k, Au \rangle + \frac{\alpha}{2} \|Au + Bz^{k+1} - b\|^2\| \leq \nu_k$$

$$3. \lambda^{k+1} = \lambda^k + \alpha(Au^{k+1} + Bz^{k+1} - b).$$

If there exists a saddle point of $L_\alpha(z, u, \lambda)$ then $(z^k, u^k, \lambda^k) \rightarrow (z^*, u^*, \lambda^*)$ which is such a saddle points. If no such saddle point exists, then at least one of the sequences $\{u^k\}$ or $\{\lambda^k\}$ is unbounded.

2.2 Decorrelation

It is well known the RGB components of a color image $u^c = (u^R, u^G, u^B)^T$ are strongly statistically correlated. It is possible to show, from an experimental point of view, (Alleyson et al. [2]), that there exists a basis $L, C^{G/M}, C^{R/B}$ in which the image $u^d = (u^L, u^{G/M}, u^{R/B})^T$ is now approximately decorrelated.

This new orthonormal basis $L, C^{G/M}, C^{R/B}$ with decorrelation is given by:

- $L = \frac{1}{\sqrt{3}}[1, 1, 1]^T$ is the luminance
- $C^{G/M} = \frac{1}{\sqrt{2}}[-1, 2, -1]^T$ is the green magenta chrominance
- $C^{R/B} = \frac{1}{\sqrt{2}}[1, 0, -1]^T$ the red blue chrominance

Moreover the change of basis matrices have the following expression:

$$(2.4) \quad u^d = \begin{bmatrix} u^L \\ u^{G/M} \\ u^{R/B} \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ -\frac{1}{4} & 0 & \frac{1}{4} \end{bmatrix} \begin{bmatrix} u^R \\ u^G \\ u^B \end{bmatrix} = T(u^c)$$

and

$$(2.5) \quad u^c = \begin{bmatrix} u^R \\ u^G \\ u^B \end{bmatrix} = \begin{bmatrix} 1 & -1 & -2 \\ 1 & 1 & 0 \\ 1 & -10 & 2 \end{bmatrix} \begin{bmatrix} u^L \\ u^{G/M} \\ u^{R/B} \end{bmatrix} = T^{-1}(u^d).$$

Hereafter u^c denotes the image in the canonical basis R, G, B , while u^d is the image in the basis $L, C^{G/M}, C^{R/B}$.

2.3 Bayer filter and blur operator

For every $(i, j) \in Z^2$ we define the color image $u^c = (u^c(i, j))_{(i, j) \in Z^2}$ where

$$u^c(i, j) = [u^R(i, j), u^G(i, j), u^B(i, j)]^T$$

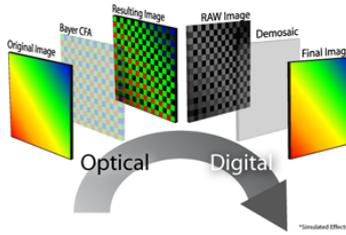


Figure 2.1: Bayer filter

is the color of the pixel of u^c at location (i, j) in the canonical R, G, B base. We define the following Bayer filter

$$(2.6) \quad u^c = [u^R(i, j), u^G(i, j), u^B(i, j)]^T \rightarrow B(u^c) = (u^c)^{X(i, j)} \quad \text{with } X(i, j) \in \{R, G, B\} \forall (i, j),$$

So that the image $(u^c)^{X(i, j)}$ has only one of the components RGB per spatial position.

Concerning the blur operator we assume that it is the same for every components. In particular we suppose the following form (with abuse of notation):

$$H = \begin{bmatrix} H & 0 & 0 \\ 0 & H & 0 \\ 0 & 0 & H \end{bmatrix}$$

Where H is a matrix representing standard convolution with some Gaussian kernel.

2.4 The variational model

Let us start by recalling the acquisition camera sequence. We have as usual:

$$(2.7) \quad u^c \mapsto Hu^c \mapsto BHu^c \rightarrow BHu^c + b = u_0.$$

On the other hand from (2.5) we have $u^c = T^{-1}(u^d)$. So that from (2.7) we get an ideal acquisition process for u^d

$$(2.8) \quad u^d \mapsto T^{-1}(u^d) \mapsto HT^{-1}(u^d) \mapsto BHT^{-1}(u^d) \rightarrow BHT^{-1}(u^d) + b = u_0.$$

The idea is then to restore u^d by working with the, much more convenient, decorrelation basis $L, C^{G/M}, C^{R/B}$. Finally, at once u^d is restored, simply set $u^c = T(u^d)$.

In order to retrieve u^d , we have to solve an ill posed inverse problem. So that as usual we seek for minimizer of an energy given by an L^2 -discrepancy term plus a regularization penalty.

Now the key point is that, since we are working in the decorrelation basis, it makes to consider the following minimization problem:

$$\arg \min_{u^d} \|\nabla u^L\|_1 + \|\nabla u^{G/M}\|_1 + \|\nabla u^{R/B}\|_1 + \mu \|BHT^{-1}(u^d) - u_0\|_2^2.$$

2.5 Application of ADMM method to our problem

In order to apply the ADMM method, we must rewrite the problem

$$(2.9) \quad \arg \min_{u^d} \|\nabla u^L\|_1 + \|\nabla u^{G/M}\|_1 + \|\nabla u^{R/B}\|_1 + \mu \|BHT^{-1}(u^d) - u_0\|_2^2,$$

in the form (2.1), which was

$$\min_{u,z} F(z) + G(u^d) \quad \text{subject to } Bz + Au^d = b.$$

Then we set

$$(2.10) \quad z = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ v \end{bmatrix} = \begin{bmatrix} \nabla u^L \\ \nabla u^{G/M} \\ \nabla u^{R/B} \\ BHT^{-1}(u^d) - u_0 \end{bmatrix}, \quad B = -I, \quad A = \begin{bmatrix} \nabla^L \\ \nabla^{G/M} \\ \nabla^{R/B} \\ BHT^{-1} \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ u_0 \end{bmatrix}$$

We also need the dual variable

$$\lambda = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ q \end{bmatrix}.$$

To simplify the notation we write

$$(2.11) \quad z = \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} \nabla u^d \\ BHT^{-1}(u^d) - u_0 \end{bmatrix} A = \begin{bmatrix} \nabla \\ K \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ u_0 \end{bmatrix}$$

an finally

$$\lambda = \begin{bmatrix} p \\ q \end{bmatrix}$$

We can now write down the corresponding augmented lagrangian as:

$$(2.12) \quad \begin{aligned} L_\alpha(z, u^d, \lambda) &= \|w\|_1 + \mu \|v\|_1^2 + \langle p, \nabla u^d - w \rangle + \langle q, Ku^d - u_0 - v \rangle \\ &+ \frac{\alpha}{2} \|v - Ku^d + u_0\|^2. \end{aligned}$$

The ADMM iterations are then given by:

$$\begin{aligned} w^{k+1} &= \operatorname{argmin}_w \|w\|_1 + \frac{\alpha}{2} \|w - \nabla(u^d)^k - D(u^d)^k - \frac{p^k}{\alpha}\|_2^2 \\ v^{k+1} &= \operatorname{argmin}_v \mu \|v\|_1 + \frac{\alpha}{2} \|v - K(u^d)^k + u_0 - \frac{q^k}{\alpha}\|_2^2 \\ (u^d)^{k+1} &= \operatorname{argmin}_u \frac{\alpha}{2} \|\nabla u^d - w^{k+1} + \frac{p^k}{\alpha}\|_2^2 + \frac{\alpha}{2} \|Ku - v^{k+1} - u_0 + \frac{q^k}{\alpha}\|_2^2 \\ p^{k+1} &= p^k + \alpha(\nabla(u^d)^{k+1} - w^{k+1}) \\ q^{k+1} &= q^k + \alpha(K(u^d)^{k+1} - u_0 - v^{k+1}), \end{aligned}$$

with $p^0 = q^0 = 0$ $\alpha > 0$.

The standard explicit formulas for w^{k+1} , v^{k+1} and $(u^d)^{k+1}$ are:

$$\begin{aligned}
 w^{k+1} &= S_{\frac{1}{\alpha}}(\nabla(u^d)^k + \frac{p^k}{\alpha}) \\
 v^{k+1} &= S_{\frac{\mu}{\alpha}}(K(u^d)^k - u_0 + \frac{q^k}{\alpha}) \\
 (2.13) \quad (u^d)^{k+1} &= (-\Delta + K^*K)^{-1}(\nabla^*w^{k+1} + K^*(v^{k+1} + u_0))
 \end{aligned}$$

where $S_{\frac{1}{\alpha}}(t)$ is the standard soft thresholding, that is

$$S_{\frac{1}{\alpha}}(t) = \begin{cases} t - \frac{1}{\alpha} \text{sign}(t) & |t| > \frac{1}{\alpha} \\ 0 & \text{otherwise.} \end{cases}$$

$S_{\frac{\mu}{\alpha}}$ is defined in the same way, up to the obvious replacement of $\frac{1}{\alpha}$ with $\frac{\mu}{\alpha}$. K^* denotes the adjoint matrix of the matrix $K = BHT^{-1}$ given by $K^* = (T^{-1})^*H^*B^*$. Note that one can compute all of these adjoint operators. Δ denotes the usual Laplace's operator.

While, concerning the last iteration of system (2.13), we used a classical conjugate gradient method.

2.6 Numerics

We test our method on images of big size (number of pixels $1550 \leq P \leq 4000$). In order to have a blurred mosaicked image to test, we follow the following standard procedure:

- 1 we pick a color image as a reference u^c , which is a good approximation of a color image to without mosaicking effect.;
- 2 we apply in the right order the acquisition operator to get the observed degraded image u_0 :

$$u_0 = BH u^c + b;$$

- 3 we formally write $u^c = T^{-1}u^d$ and we work with the new basis $(u^L, u^{\frac{G}{M}}, u^{\frac{R}{B}})$. So we have

$$u_0 = BHT^{-1}(u^d) + b;$$

- 4 We apply the ADMM algorithm to restore u^d ;
- 5 We set $u^c = T(u^d)$.

As blur operator we always have considered a standard Gaussian low pass filter of size $h = 11$, with standard deviation $\epsilon = 1$. In figures 2.2 2.3,2.4 we restore an image of size 2.200 with a low level of noise. When the level noise is high, μ cannot be too small otherwise, the algorithm does not perform a good demosaicking. In this case the parameter μ is chosen in order to have

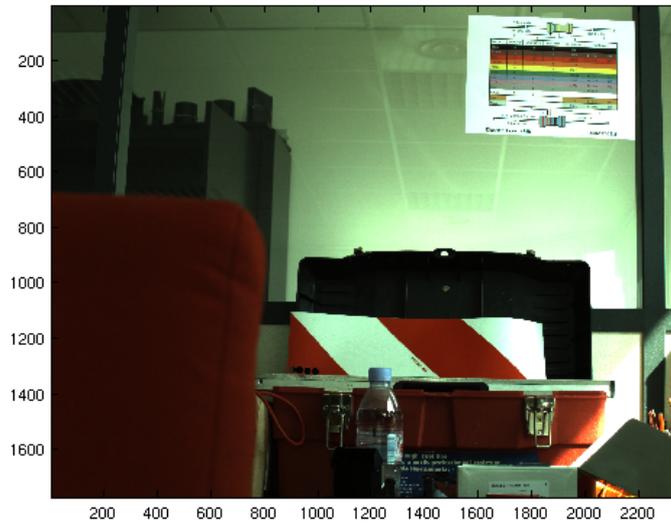


Figure 2.2: Original image $u^c = T^{-1}(u^d)$. Size image 2200x2000

a good balancing between denoising and demosaicking. In figure 2.5 we show the restoration results of an image reference detail with different value of the parameter μ . Then in figures 2.7,2.8 we show the restoration result obtained on the whole image.

We deal with rescaled images in $[0, 1]$. We made run the Matlab code on an Intel(R) Xeon(R) CPU 5120 @ 1.86GHz.

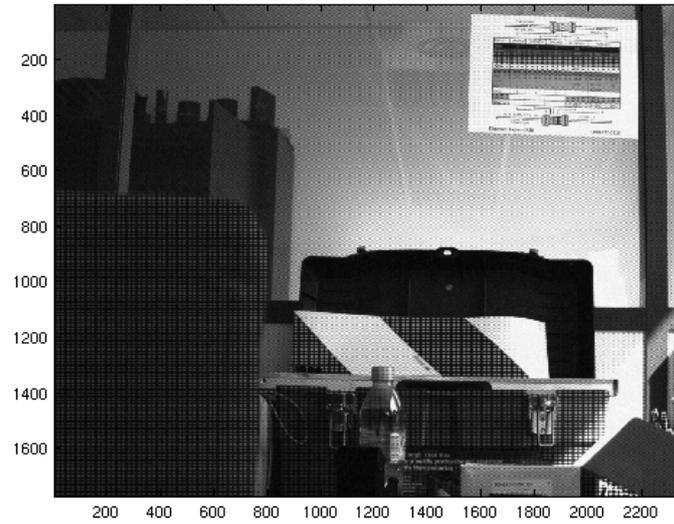


Figure 2.3: Observed mosaicked blurred and noisy image $u_0 = BHT^{-1}(u^d) + b$. $\sigma = 0.01$.

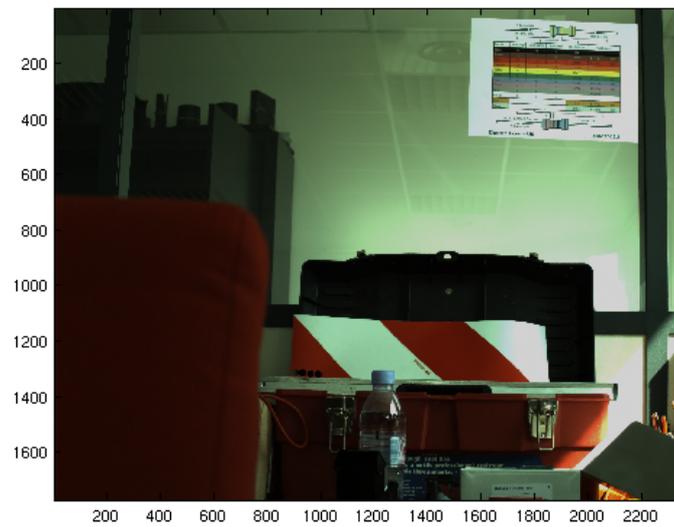


Figure 2.4: Restored image $u^c = T(u^d)$. CPU time about 30mn, number of iterations 30 $\mu = 30$

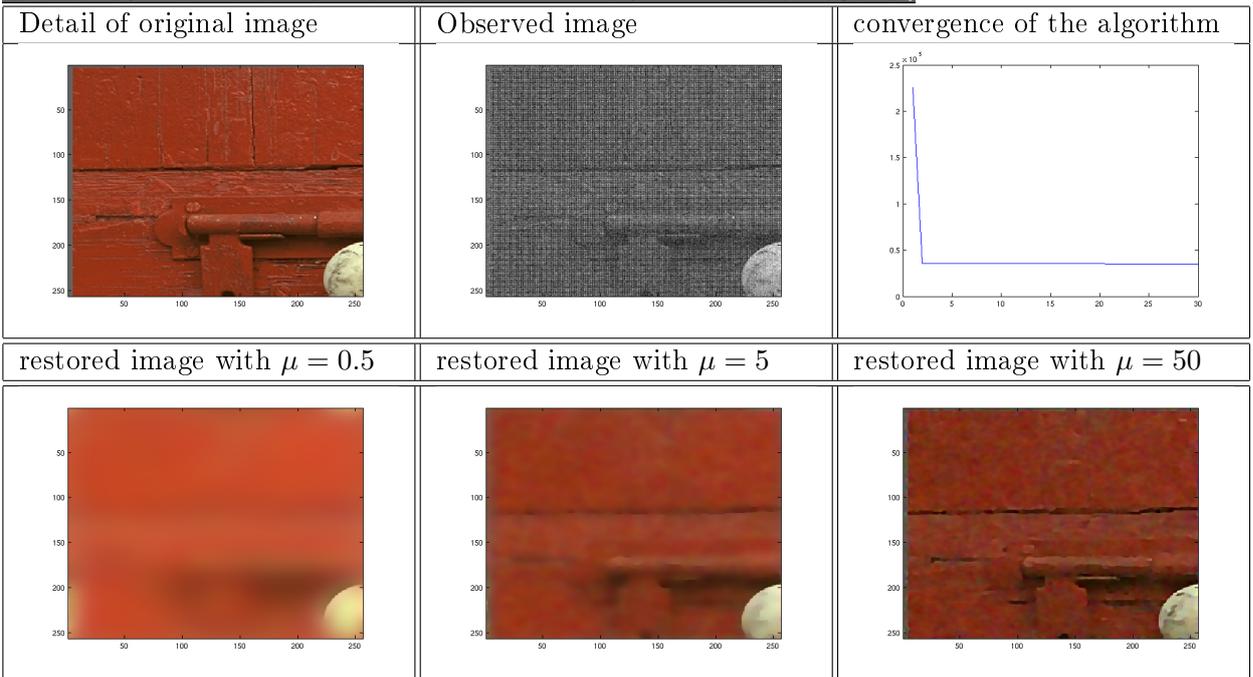


Figure 2.5: Top left: crop of the original image. Crop size 256x256. Top center: blurred mosaicked noisy image. Top right: convergence of the algorithm. Down left: restored image with a small μ to promote the denoising against the demosaicking. Number of iterations 30. Down center and down right : restored image with a greater value of μ to promote demosaicking against the denoising. Number of iterations 30. Cpu time 4mn.



Figure 2.6: Original image of size 768x512

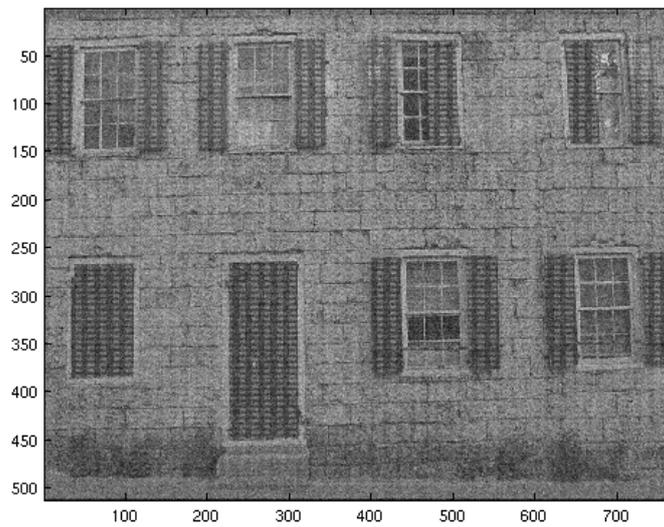


Figure 2.7: Observed mosaicked blurred and noisy image. $\sigma = 0.5$



Figure 2.8: Restored image $u^c = T(u^d)$. CPU time about 20 mn, number of iterations 30
 $\mu = 20$

Acknowledgments: The first author author warmly thanks Mikael Carlván, Phd student of Morpheme project at I3s, to help him with the Matlab code of the ADMM algorithm.

Bibliography

- [1] D.Alleyson *30 ans de démosaïcage* Traitement du signal vol 21 2000, 563-581
- [2] D. Alleyson, S.Susstrunk and J.Hérault *Linear demosaicking inspired by the human visual system* IEEE Trans. Image Processing, vol.14, no.4 pp.439-449. Apr. 2005
- [3] A.Chambolle *An algorithm for total variation minimization and applications* J. Math. Imaging Vision 20 (2004), no. 1-2, 89–97.
- [4] L.Condat *A simple fast and efficient approach to demosaicking and denoising* Proceeding of 2010 IEEE 17th International conference on Image processing Hong Kong.
- [5] J.Eckstein D.Bertsekas *On the Douglas-Rachford splitting method and proximal monotone operators*, Mathematical Programming 55, North-Holland 1992
- [6] E. Essier *Primal Dual Algorithms for Convex Models and Applications to Image Restoration, Registration and Nonlocal Inpainting* Phd Thesis 2009.
- [7] S. Farsiu, M. Eladb , P. Milanfar *Multi-Frame Demosaicing and Super-Resolution from Under-Sampled Color Images* SPIE Symposium on Electronic Imaging 2004, January 2004 in San Jose, CA. Volume 5299, Page(s): 222-233.
- [8] S.Farsiu, M.Robinson,M.Elad, P.Milanfar *Fast and Robust Multiframe Super Resolution* IEEE Transactions on Image Processing Volume: 13 n.10 pp.1327-1344 Octobre 2004
- [9] M.Fornasier, A. Langer, C. Schonlieb. *A convergent overlapping decomposition method for total variation Minimization".* Numerische Mathematik, Vol. 116, No. 4, 2010, pp. 645-685
- [10] M.Fornasier, C. Schonlieb *Subspace correction method for total variation minimization*, Siam J. Numer. Anal., Vol. 47, No. 5, 2009, pp. 3397-3428
- [11] Y.Nesterov. *A method for unconstrained convex minimization problem with the rate of convergence $O(\frac{1}{k^2})$.* Doklady AN SSSR **547** (1983), 543-547.
- [12] Y. Nesterov. *Smooth minimization of non-smooth functions.* Mathematic Programming, Ser. A, **103** (2005) ,127-152.

- [13] P.Weiss. *Algorithmes rapides d'optimisation convexe. Application à la restauration d'images et à la détection de changements*. Phd Thesis (2008).
- [14] P. Weiss, L. Blanc-Féraud , G. Aubert. *Efficient schemes for total variation minimization under constraints in image processing*. SIAM journal on Scientific Computing **31** (2009), 2047-2080.
- [15] F. Soulez, E. Thiébaud *Joint deconvolution and demosaicing* 16th International conference on image processing Cairo Egypt.
- [16] M. Unger, T. Pock, M. Werlberger, H. Bischof *A convex approach for variational super-resolution*, Proceedings of the 32nd DAGM conference on Pattern recognition.